REMARKS

Reconsideration and allowance of the present application are respectfully requested. Claims 1, 4-19, 21-27, 29-31, and 33-42 are currently pending in this application.

Regarding the 35 U.S.C. § 101 Rejection

The Office Action again rejects claims 1, 2 and 4-7 under 35 U.S.C. § 101 because the claimed invention is alleged to be directed to non-statutory subject matter. More specifically, in paragraph No. 8, the Office Action alleges that the claims 1-7 "consist solely of computer program, which is non-statutory functional descriptive material." The Office Action further states that the "language of the claims does not recite any computer hardware involvement." The Applicant respectfully traverses this rejection for the following reasons. (As claim 3 has been canceled herein, the rejection is discussed with respect to the remaining claims, i.e., claims 1 and 4-7.)

Claims 1 and 4-7 are directed to, in part, a "*system*" comprising "*a pluggable security policy enforcement module* configured to be replaceable in the system and to provide different granularities of control for *a business logic in the system*, wherein the business logic *processes* requests submitted to the system." The terms "system," "pluggable security policy enforcement module," and "business logic" all point to physical mechanisms for implementing the invention, rather than a mere descriptive recitation of a program *per se*. For example, by virtue of the fact that "business logic" is recited which "processes requests," these claims refer to a physical agent which performs the processing, rather than a mere description of what the processing entails. Claims 1 and 4-7 should therefore be classified as statutory product claims.

In fact, the MPEP itself states, in discussing functional descriptive material, that, "When a computer program is recited in conjunction with a physical structure, such as a computer memory, Office personnel should treat the claim as a product claim," (MPEP § 2106, page 2100-13 of the May 2004 revision). It is true that *one* exemplary and non-limiting way of implementing the pluggable security policy enforcement module is using software; however, in accordance with the MPEP, software is not being claimed *per se* in a proscribed descriptive manner. The Office Action cites no authority to counter the express instructions of the MPEP; namely, the Office Action cites no authority for its position that a *system* claim may be construed as functional descriptive material.

For the above-identified reasons, the Applicant respectfully requests that the rejection of claims 1 and 4-7 be withdrawn. In the alternative, if this rejection is repeated, the Patent Office is respectfully requested to support its position by citing the authority it is relying on.

Regarding the 35 U.S.C. § 102 Rejection

All of the claims, i.e., claim 1, 2, and 4-42 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,820,082 to Cook et al. (referred to below as "Cook"). Applicant respectfully traverses this rejection for the following reasons. (As claims 2, 3, 20, 28, and 32 have been canceled herein, the rejection is discussed with respect to the remaining claims, i.e., claims 1, 4-19, 21-27, 29-31, and 33-42.)

Cook's invention is directed to a rule-based database security system and method. As shown in Fig. 1, Cook's discloses a system that includes an application server 70 configured to interface with a client (user) browser 72 and a database 74. The "application server 70 includes a user interface 76, a rules engine (business logic) 78, and a data access manager 80" (column 4, lines 54-56). The user interface 76, rules engine

78, and database 74 are configured so that each may be changed and maintained independently (column 4, lines 61-63). More specifically, the system 10 utilizes rule-based business logic, rather than object oriented code based business objects, which allows for modification to the security of the database, either for viewing or updating, without modifying code and re-compiling the system software (column 6, lines 38-42).

Each rule includes a name, an expression, a description, and possibly a help ID which points to information about what may make a rule fail, for example (column 6, lines 48-51). The rules may be grouped into four categories: transaction control; action triggering; object initialization; and access control. Transaction control rules govern what kind of changes a user can make to the database. The trigger rules activate the system and are configured to respond to changes in the state of the system. Object initialization rules govern the operations which occur upon initialization. Access control rules govern access to information. See column 6, line 46 to column 7, line 15.

Claim 1 has been amended by incorporating the subject matter of claim 2, and by clarifying the use of the term "business logic." This claim is reproduced below with emphasis added:

> 1. A system comprising:
>
> a pluggable security policy enforcement module configured to be replaceable in the system and to provide different granularities of control for a business logic in the system, wherein the business logic processes requests submitted to the system, *wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain,*
>
> wherein the pluggable security policy enforcement module is configured to determine, for a particular granularity of control, whether to permit an operation, requested by a user based at least in part on a permission assigned to the user,

and wherein the different granularities of control comprise a plurality of sets of rules that can be replaced with each other without altering the business logic.

Cook fails to teach at least the clause which recites, "wherein the different granularities of control comprise a plurality of sets of rules that can be replaced with each other without altering the business logic," in combination with the other elements of the claim, when considered as a whole. More specifically, as discussed above, Cook provides a set of rules, but these rules are considered synonymous with what Cook refers to as business logic. For instance, in column 4, line 55, Cook refers to rule engine (business logic) 78," indicating that the rule engine 78 is coextensive with the business logic. As a consequence, Cook's set of rules cannot be replaced "without altering the business logic," as recited in the terminal clause of claim 1. This is because, in Cook, the rules are not separate from what Cook is referring to as business logic. Moreover, there is no indication that the mechanism which Cook is referring to as business logic contains problem-solving logic that produces solutions for a particular problem domain, as now expressly recited in claim 1.

In rejecting this subject matter, the Office Action identifies column 2, lines 15-54 of Cook as having relevance to the clause in question (note paragraph No. 12 of the Office Action). This passage provides an overview of Cook's technique, but does not disclose the concept of a plurality of sets of rules that can be replaced with each other without altering the business logic.

Cook fails to disclose or suggest all of the elements in claim 1 for at least the above-identified reasons.

Advancing to independent claim 4, this claim has been amended by clarifying the use of the term "business logic." This claim is reproduced below with emphasis added:

4. (Currently amended) A system comprising:

a pluggable security policy enforcement module configured to be replaceable in the system and to provide different granularities of control for a business logic in the system, wherein the business logic processes requests submitted to the system, *wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain,*

wherein the pluggable security policy enforcement module includes a control module configured to determine whether to permit an operation based at least in part on accessing the business logic to identify one or more additional tests to perform, and further configured to perform the one or more additional tests.

Cook fails to teach at least the clause which recites, "a control module configured to determine whether to permit an operation based at least in part on accessing the business logic to identify one or more additional tests to perform, and further configured to perform the one or more additional tests," in combination with the other elements of the claim, when considered as a whole. More specifically, Cook discloses a rules engine 78, which applies a number of security-related rules. But again, Cook's rules engine 78 is coextensive with what Cook refers to as its business logic. Therefore Cook cannot be said to provide a pluggable security policy enforcement module and *also* rely on the business logic to "identify one or more *additional* tests to perform." In other words, in Cook, the rules engine 78 is the locus of security procedures, so there is no disclosed provision for accessing any other entity to determine whether *additional* tests should be performed.

It is noted that Cook's access manager 86 (within the rules engine 78) modifies a request to produce a modified query. Then, when the results are obtained based on this

modified query, the access manager 86 can apply field level access control to further filter the results, e.g., by removing information that is not available to the user. Note column 10, line 64 to column 11, line 12 of Cook. But this mechanism refers to processing performed within the rules engine 78. This mechanism does not involve accessing separate business logic to identify one or more additional tests to perform. Moreover, to further clarify the differences between Cook and the invention recited in claim 4, claim 4 has been amended to state that the "business logic contains problem-solving logic that produces solutions for a particular problem domain." Cook's modifying of the query and filtering of the results (as described above) do not involve accessing business logic that is defined in the manner recited in claim 4.

In rejecting this subject matter, the Office Action identifies various excerpts of columns 1 and 2 of Cook as having relevance to the clause in question (note paragraph No. 11 of the Office Action). However, while this portion mentions rules-based processing and business logic, there is no disclosure in this portion that separate business logic is relied on to identify one or more additional tests to perform. In any event, it is noted that paragraph No. 11 does not specifically address the clause identified above in claim 4. If this rejection is repeated, the Patent Office is respectfully requested to point out the passage in Cook that it is being relied on to reject the clause in question.

Cook fails to disclose or suggest all of the elements in claim 4 for at least the above-identified reasons.

Advancing to independent claim 6, this claim is reproduced below with emphasis added:

> 6. A system comprising:

a pluggable security policy enforcement module configured to be replaceable in the system and to provide different granularities of control for a business logic in the system, wherein the business logic processes requests submitted to the system,

wherein the different granularities of control comprise a plurality of sets of rules, and wherein each set of rules includes a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed.

Cook fails to teach at least the clause which recites that, "the different granularities of control comprise a plurality of sets of rules, and wherein each set of rules includes a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed," in combination with the other elements of the claim, when considered as a whole. More specifically, Cook describes a system in which each rule includes a name, an expression, a description, and possibly a help ID which points to information about what may make a rule fail. The rules may be grouped into four categories: transaction control; action triggering; object initialization; and access control (column 6, lines 44-51). However, claim 4 recites a specific *organization* of information that defines the rules, wherein each set of rules includes a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and

wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed. Cook does not describe this specific organization of information.

In rejecting this subject matter, the Office Action identifies column 7, line 8 to column 9, line 60 of Cook as having relevance to the clause in question (in paragraph No. 13 of the Office Action). This portion of Cook discusses, in part, Cook's access control rules. Access control rules govern who can see what information (column 7, lines 8-9). But these rules are not *structured* in the specific manner recited in claim 6. In other words, claim 6 does not simply recite a laundry list of security-related features, but a specific organization of such features; Cook does not disclose or even hint at this subject matter.

Cook fails to disclose or suggest all of the elements in claim 6 for at least the above-identified reasons.

Advancing to independent claim 8, this claim has been amended by clarifying the use of the term "business logic." This claim is reproduced below with emphasis added:

> 8. (Currently amended) One or more computer-readable media comprising computer-executable instructions that, when executed, direct a processor to perform acts including:
>
> receiving a request to perform an operation;
>
> *checking whether to access a business logic in order to generate a result for the requested operation, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain;*
>
> *obtaining, from the business logic, a set of zero or more additional tests to be performed in order to generate the result;*

performing each additional test in the set of tests if there is at least one test in the set of tests;

checking a set of pluggable rules to determine the result of the requested operation; and

returning, as the result, a failure indication if checking the business logic or checking the set of pluggable rules indicates that the result is a failure, otherwise returning, as the result, a success indication.

Cook fails to teach at least the clauses which recite, "checking whether to access a business logic in order to generate a result for the requested operation," "obtaining, from the business logic, a set of zero or more additional tests to be performed in order to generate the result," and "performing each additional test in the set of tests if there is at least one test in the set of tests," in combination with the other elements of the claim, when considered as a whole. As stated with respect to claim 4, Cook discloses a rules engine 78, which applies a number of security-related rules. But again, Cook's rules engine 78 is coextensive with what Cook refers to as its business logic. Therefore, Cook cannot be said to obtain, "from the business logic, a set of zero or more additional tests to be performed in order to generate the result." That is, in Cook, the rules engine 78 is the locus of security procedures, so there is no disclosed provision for accessing any other entity to determine whether *additional* tests should be performed.

In rejecting this subject matter, the Office Action identifies column 5, lines 29-41 as having relevance to the clauses in question (in paragraph No. 16 of the Office Action). This passage is reproduced as follows:

The request is processed by loading appropriate templates, which specify the type and format of information to be returned for various kinds of information requests. The templates, together with

> the original request, specify what information is needed from the database to satisfy the request. The required information is formulated as a query (FIG. 1), which specifies what information is needed. The query is passed to the rule engine 78 so that only the information that is appropriate for the current user, based on applicable security constraints, is returned. The user interface 76 also includes a page generator 84 for generating a page with the results of the query or update, or error notification for display to the user by the browser 72.

This passage merely recites a procedure performed by the user interface 76 for supplementing a request before it is submitted to the rule engine 78. This passage does not describe checking whether to access business logic in order to generate a result for the requested operation, obtaining, from the business logic, a set of zero or more additional tests to be performed in order to generate the result, and performing each additional test in the set of tests if there is at least one test in the set of tests. Moreover, to further preclude interpretation of, say, the template-based processing performed by the user interface 76 as "business logic," claim 8 has been amended to recite that "the business logic contains problem-solving logic that produces solutions for a particular problem domain." Cook's user interface 76 cannot be characterized in this manner.

Cook fails to disclose or suggest all of the elements in claim 8 for at least the above-identified reasons.

Advancing to independent claim 19, this claim has been amended by incorporating the subject matter of claim 20, and by clarifying the use of the term "business logic." This claim is reproduced below with emphasis added:

> 19. (Currently amended) A method comprising:
>
> providing high-level permission concepts for security rules;

allowing a set of security rules to be defined using the high-level permission concepts, wherein the set of security rules allows permissions to be assigned to users of an application; and

determining, based at least in part on a permission assigned to a user, whether to permit an operation based on a request by the user,

wherein the determining further comprises determining whether to permit the operation requested by the user based at least in part on accessing a business logic to identify one or more additional tests to perform, and further comprising performing the one or more additional tests, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain.

For reasons similar to those presented for claim 8, Cook fails to teach at least the clause which recites, "wherein the determining further comprises determining whether to permit the operation requested by the user based at least in part on accessing a business logic to identify one or more additional tests to perform, and further comprising performing the one or more additional tests, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain," in combination with the other elements of the claim, when considered as a whole.

In rejecting this subject matter, the Office Action identifies column 6, lines 38-54 of Cook as having relevance to the clause in question (paragraph No. 21 of the Office Action). This passage mentions the term business logic (e.g., "As described above, the system 10 utilizes rule-based business logic, rather than object oriented code based business objects", in column 6, lines 38-40). But, as stated above, Cook's rule-based processing is not separate from what Cook is referring to as business logic. Thus, Cook does not disclose an operation for accessing business logic to identify one or more

additional tests to perform, where the business logic is defined in manner recited in claim 19.

Cook fails to disclose or suggest all of the elements in claim 19 for at least the above-identified reasons.

Advancing to independent claim 26, this claim has been amended by incorporating the subject matter of claim 28, and by clarifying the use of the term "business logic." This claim is reproduced below with emphasis added:

> 26. (Currently amended) A method comprising:
>
> receiving a request to perform an operation associated with business logic, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain;
>
> accessing a set of low-level rules, wherein the low-level rules are defined in terms of high-level concepts;
>
> checking whether a user requesting to perform the operation is entitled to perform the operation based at least in part on the set of low-level rules; and
>
> returning an indication of whether the operation is allowed or not allowed,
>
> wherein the set of low-level rules can be replaced with another set of low-level rules without altering the business logic.

For reasons similar to those given for claim 1, Cook fails to teach at least the clause which recites, "wherein the set of low-level rules can be replaced with another set of low-level rules without altering the business logic," in combination with the other elements of the claim, when considered as a whole.

In rejecting this subject matter, the Office Action identifies column 2, lines 15-54 of Cook as having some bearing on the clause in question (paragraph No. 12 of the Office Action). As previously stated, this portion provides an overview of Cook's technique, but does not disclose the concept of a plurality of sets of rules that can be replaced with each other without altering the business logic,

Cook fails to disclose or suggest all of the elements in claim 26 for at least the above-identified reasons.

Advancing to independent claim 31, this claim has been amended by incorporating the subject matter of claim 32, and by clarifying the use of the term "business logic." This claim is reproduced below with emphasis added:

> 31. A method comprising:
>
> assigning high level security concepts to an application domain; and
>
> allowing a set of pluggable rules to define low-level rules, in terms of the high level security concepts, for different business logic in the application domain, *wherein each business logic contains problem-solving logic that produces solutions for a particular problem domain,*
>
> *wherein the high level security concepts include an operation that identifies an operation to be performed, and a context that identifies what the operation is performed on.*

Cook fails to teach at least the clause which recites, "allowing a set of pluggable rules to define low-level rules, in terms of the high level security concepts, for different business logic in the application domain, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain, wherein the high level security concepts include an operation that identifies an operation to be performed, and a context that identifies what the operation is performed on," in combination with the

other elements of the claim, when considered as a whole. More specifically, Cook describes a system in which each rule includes a name, an expression, a description, and possibly a help ID which points to information about what may make a rule fail. The rules may be grouped into four categories: transaction control; action triggering; object initialization; and access control (column 6, lines 46-51). But Cook does not describe allowing a set of security rules to be defined using high-level permission concepts, "wherein the high level security concepts include an operation that identifies an operation to be performed, and a context that identifies what the operation is performed on." That is, while Cook discusses the rules in generalities, which the Examiner may be interpreting as high-level permission concepts, the generalities identified by Cook do not conform to the rubric of "operation" and "context," as recited in claim 31. Nor does Cook describe allowing high level security concepts to be defined for different business logic, where the business logic is defined in the manner now claimed.

In rejecting this subject matter, the Office Action identifies column 6, lines 11-30 as having some bearing on the clause in question (paragraph No. 18 of the Office Action). This passage describes various features of transaction control rules, but it does not describe the above-identified structure of the high level security concepts (involving "operation" and "context"). This passage also does not describe the claimed manner in which high level security concepts are defined for different business logic.

Cook fails to disclose or suggest all of the elements in claim 31 for at least the above-identified reasons.

Advancing to independent claim 35, this claim has been amended by clarifying the use of the term "business logic" and by clarifying that the pluggable security policy enforcement module is separate from the business logic. This claim is reproduced below with emphasis added:

35. An architecture comprising:

a plurality of resources;

a business logic layer to process, based at least in part on the plurality of resources, requests received from a client, *wherein the business logic layer contains problem-solving logic that produces solutions for a particular problem domain*; and

a pluggable security policy enforcement module, *separate from the business logic layer*, to enforce security restrictions on accessing information stored at the plurality of resources.

Cook fails to teach at least the clause which recites, "a pluggable security policy enforcement module, separate from the business logic layer, to enforce security restrictions on accessing information stored at the plurality of resources," in combination with the other elements of the claim, when considered as a whole. More specifically, as described above, Cook does not describe a pluggable security policy enforcement module which is separate from business logic (as defined in claim 35), because Cook's rule engine 78 appears to be coextensive with what Cook is referring to as business logic.

In rejecting this subject matter, the Office Action identifies column 7, line 8 to column 8, line 61 of Cook (paragraph No. 25 of the Office Action). This passage of Cook describes various kinds of access control rules, and also describes the manner in which the rule engine modifies a request. This passage does not, however, disclose a pluggable security policy enforcement module that is separate from a business logic layer.

Cook fails to disclose or suggest all of the elements in claim 35 for at least the above-identified reasons.

The remaining pending claims are dependent claims. Since Cook fails to disclose all of the elements in the independent claims, Cook fails to disclose all of the elements in each of the claims which depend from these independent claims. In addition, the dependent claims recite various additional features not disclosed by Cook.

For at least the above reasons, the Applicant submits that Cook does not anticipate any of the claims. Namely, as stated in MPEP § 2131, a claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California*, 2 USPQ2d 1051 (Fed. Cir. 1987). Since Cook does not set forth each and every feature of the claims, it fails to anticipate the claims under § 102.

For the above-identified reasons, the Applicant respectfully requests that the 35 U.S.C. § 102(e) rejection be withdrawn.

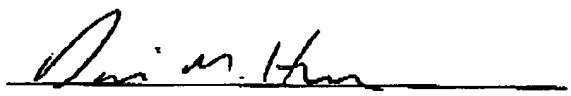
Conclusion

The arguments presented above are not exhaustive; Applicant reserves the right to present additional arguments to fortify its position. Further, Applicant reserves the right to challenge the alleged prior art status of one or more documents cited in the Office Action.

All objections and rejections raised in the Office Action having been addressed, it is respectfully submitted that the present application is in condition for allowance and such allowance is respectfully solicited. The Examiner is urged to contact the undersigned if any issues remain unresolved by this Amendment.

Respectfully Submitted,

Dated: 10-3-2005

By: 

David M. Huntley
Reg. No. 40,309
(509) 324-9256